Definition of MeasurementSet
AIPS++ Note 191

M.H. Wieringa, ATNF
T.J. Cornwell, NRAO

October 4, 1996

Contents

1 Summary

A MeasurementSet is an AIPS++ Table containing data from a telescope. Telescopes all record data in their own way, the MeasurementSet layout describes how these measurements are to be stored within the AIPS++ system. This document describes the predefined columns and keywords (including subtables) of the MeasurementSet (MS). The predefined items include required and optional ones. In addition to the predefined columns, MeasurementSets can contain instrument specific columns, keywords or subtables. These additional items would only be accessed from instrument specific code.

2 Details

- The purpose of the columns and tables described here is to store information that one can expect to arrive from a telescope. Calibrated data will be the DATA column in a calibrated MS that is otherwise simply a reference to the original MS (note that the UNITs of the DATA column may therefore be different in original and calibrated MSes). Predicted data will similarly show up as a DATA column in an MS.
- There is only one subtable of each type and that is stored as a keyword of the table. This simplifies selection since it can be done on a table (e.g. SOURCE NAME is '3C84') and the keys then used to select from the main table.
- We want to keep as many columns as possible common to both Single Dish and Synthesis. Furthermore, we regard most of the columns identified here as required. It simplifies code tremendously and the Miriad storage manager avoids significant disk space usage. However, we have identified a couple of cases where following these guidelines would result in too much wasted storage and introduced some optional columns to cope with this (e.g., FLOAT_DATA for single dish).
- The model is that each ANTENNA has an unrestricted number of FEEDs. For each FEED there are some RECEPTORS, probably one per polarization state (e.g. R and L or X and Y). A number of SPECTRAL_WINDOW setups are allowed. Different setups are expected in different rows of the Main table. Thus, for example, different VLA IFs show up as different rows.
- We think that PHASED_ANTENNAS and PHASED_FEEDS table will be needed but since these are quite specialized, we have refrained from specifying contents. We think that these should be implemented first as TELESCOPE specific tables (e.g. NRAO_GBT_FEEDARRAY) and then a common format agreed upon. Such tables should probably index into the ANTENNA and FEED tables to get information.

- The SOURCE table contains information that is often not present in on-line formats, therefore we have made the use of the table optional. It could be added during later processing to organize the data by source. The FIELD table is the place to describe pointings.
- The CORRELATOR table is still unspecified. Instruments like the VLBA will require it, but at present it will usually be empty.
- The PULSAR table appears to be very telescope specific so we haven't described any format. Perhaps this possibility would provide a good impetus for the pulsar community to make common cause.
- Non-standard information may be added in various ways as per Bob Garwood's proposal:
  - As a new column in an existing table e.g. NS_NRAO_GBT_WHATEVER
  - As an e.g. NRAO-approved column NRAO_GBT_WHATEVER or NRAO_WHATEVER
  - As a Project approved column WHATEVER.
- Units and Measures. All columns representing physical quantities should be specified in the correct units, as specified in the table descriptions. Columns representing AIPS++ Measures should have a MEASURE keyword giving the AIPS++ Measure name minus the initial 'M' (e.g., DIRECTION or EPOCH) and a MEASURE_REFERENCE keyword giving the Measure enum type as a String (e.g., "J2000" or "UTC")

3 MeasurementSet classes

The MeasurementSet is coded as a series of related classes. Each subtable follows the same basic scheme as the MAIN table. The namespace in each table is separate. For each (sub)table there is a separate class which provides various conversion and inquiry functions for the columns and keywords and defines the required set of columns and keywords. The main MeasumentSet class provides access to the subtables via these classes. E.g., MeasurementSet has a member function antenna() which gives access to the ANTENNA subtable. More detailed documentation on these classes can be found in the on-line documentation (directory: code/trial/implement/MeasurementSet).

The definite descriptions of the MS tables are contained within the AIPS++ class documentation files in the MeasurementSet module. For each of the MeasurementSet tables there is a file (e.g. MSMainEnums.html, MSAntennaEnums.html) giving a description for the enum name (e.g., DATA, POSITION) used for a particular column or keyword. Examples on how to use these classes to write a filler for a particular instrument can be found by looking at existing fillers, e.g., uvfitsfiller.

4 MS layout

There is a MAIN table containing a number of data columns and keys into various subtables. There is at most one of each subtable. The subtables are stored as keywords of the MS.

Subtables		
Table	Contents	Keys
ANTENNA	Antenna characteristics	ANTENNA_ID, ARRAY_ID
ARRAY	Array characteristics	ARRAY_ID
CORRELATOR	Correlator setup	CORRELATOR_ID
FEED	Feed characteristics	FEED_ID, ANTENNA_ID, ARRAY_ID, TIME, SPECTRAL_WINDOW_ID
FIELD	Position etc for each pointing.	FIELD_ID
FLAGGING_LOG	Log of flagging operations	ANTENNA_ID, ARRAY_ID, FEED_ID, SPECTRAL_WINDOW_ID, FIELD_ID, TIME, INTERVAL
OBSERVATION	Observer, Schedule, etc	OBSERVATION_ID
OBS_LOG	Log from on-line system	OBSERVATION_ID, TIME
SOURCE	Positions, etc for each source	SOURCE_ID, SPECTRAL_WINDOW_ID, TIME
SPECTRAL_WINDOW	Spectral window setups	SPECTRAL_WINDOW_ID
SYSCAL	System calibration characteristics	FEED_ID, ANTENNA_ID, ARRAY_ID, TIME, SPECTRAL_WINDOW_ID
WEATHER	Weather info for each antenna	ANTENNA_ID, ARRAY_ID, TIME

Note that there are two types of subtables. For the first, simpler type, the key (ID) is the row number in the subtable. Examples are ARRAY, FIELD, SPECTRAL_WINDOW, OBSERVATION and CORRELATOR. For the second, the key is a collection of parameters, usually including TIME. Examples are ANTENNA, FEED, SOURCE, SYSCAL, WEATHER. We think the Calabretta interpolation test is a good one: CORRELATOR setups cannot be sensibly interpolated and thus a different ID is required for each setup. For the interpolable tables, one needs to decide what value is actually to be used. We think this belongs in access routines independent of the MS.

Notes:

- All ID columns are zero-based, a value of -1 indicates that there is no corresponding subtable present.
- All required columns should be filled with suitable defaults if not actually used.
- For time dependent tables, a value of DBL_MAX (defined by including aips/Mathematics/Constants.h) for interval can be used to specify non time dependent entries.
- The layout of the CORRELATOR and FLAGGING_LOG table is currently unspecified. We need to gain some experience with specific implementations of these before specifying a general format.

4.1 MAIN table: Coordinates, Data, pointers and Flags

MAIN table: Coordinates, Data, pointers and Flags				
Name	Format	Units	Measure	Comments
Columns				
Coordinate information				
ANTENNA1	Int			First antenna
ANTENNA2	Int			Second antenna
EXPOSURE	Double	s		The effective integration time
FEED1	Int			Feed on ANTENNA1
FEED2	Int			Feed on ANTENNA2
INTERVAL	Double	s		Sampling interval
PULSAR_BIN	Int			Pulsar bin number
SCAN_NUMBER	Int			Scan Number
TIME	Double	s	EPOCH	Midpoint of the integration.
TIME_EXTRA_PREC	Double	s		add to TIME for extra precision (optional)
UVW	Double(3)	m	UVW	UVW coordinates
Pointers into subtables				
ARRAY_ID	Int			
CORRELATOR_ID	Int			
FIELD_ID	Int			
OBSERVATION_ID	Int			
PULSAR_ID	Int			
SPECTRAL_WINDOW_ID	Int			
Data				
DATA	Complex(NUM_CORR, NUM_CHAN)			Complex visibility matrix (synthesis arrays, optional)
FLOAT_DATA	Float(NUM_CORR, NUM_CHAN)			Float data matrix (single dish, optional)
SIGMA	Float(NUM_CORR)			Estimated rms noise for single channel
SIGMA_SPECTRUM	Float(NUM_CORR, NUM_CHAN)			Estimated rms noise (optional)
WEIGHT	Float			Weight for whole data matrix
WEIGHT_SPECTRUM	Float(NUM_CHAN)			Weight for each channel (optional)
Flag information				
FLAG	Bool(NUM_CORR, NUM_CHAN)			The data flags
FLAG_HISTORY	Bool (NUM_CORR, NUM_CHAN, NUM_HIS)			History of flags (optional)
FLAG_ROW	Bool			The row flag

Notes:

DATA, FLOAT_DATA At least one of these columns should be present in any MeasurementSet. In special cases both could be present (e.g., single dish data used in synthesis imaging or a mix of auto and crosscorrelations on a multi-feed single dish)

ANTENNA1, ANTENNA2 For Single dish ANTENNA1=ANTENNA2

PULSAR_BIN For a pulsar the correlations are assumed to be measured for a limited number of pulse phase bins. This is the particular bin for which this data was measured.

FLAG An array of bools with the same shape as DATA. Data is flagged bad if FLAG is True.

FLAG_HISTORY A set of NUM_HIS flags for each data point, each for a different category of flagging (e.g. on-line, calibration, interactive, etc.)

Next we describe the various subtables. The columns are categorized into Keys, Data description, Data, and Flags.

4.2 ANTENNA: antenna characteristics

ANTENNA: antenna characteristics				
Name	Format	Units	Measure	Comments
Columns				
Coordinate information				
ANTENNA_ID	Int			Key
ARRAY_ID	Int			Key
Pointers				
ORBIT_ID	Int			Orbit parameters
PHASED_ARRAY_ID	Int			Phased array
Data				
DISH_DIAMETER	Double	m		Diameter of dish
MOUNT	String			Mount type:alt-az, equatorial, X-Y, orbiting, bizarre
NAME	String			Antenna name
OFFSET	Double(3)	m	POSITION	Axes offset of mount to FEED REFERENCE point
POSITION	Double(3)	m	POSITION	Antenna X,Y,Z phase reference positions
STATION	String			Station name

Notes:

POSITION In a right-handed frame, X towards the intersection of the equator and the Greenwich meridian, Z towards the pole. The exact frame should be specified in the MEASURE_REFERENCE keyword (ITRF or WGS84). The reference point is the point on the az or ha axis closest to the el or dec axis.

4.3 ARRAY: Array characteristics

ARRAY: Array characteristics				
Name	Format	Units	Measure	Comments
Data				
NAME	String			Name of array (e.g. 'VLAA', 'EVN', 'VLBA-PT'). Informational only
POSITION	Double(3)	m	POSITION	Array reference position (optional), same frame as antenna positions.

4.4 FEED: Feed characteristics

FEED: Feed characteristics				
Name	Format	Units	Measure	Comments
Columns				
Keys				
ANTENNA_ID	Int			Antenna id
ARRAY_ID	Int			Array id
FEED_ID	Int			Feed id
INTERVAL	Double	s		Interval
SPECTRAL_WINDOW_ID	Int			Spectral Window id
TIME	Double	s	EPOCH	Midpoint of validity range of parameters
Data description				
NUM_RECEPTORS	Int			# receptors on this feed
Data				
BEAM_ID	Int			Beam model
BEAM_OFFSET	Double(2, NUM_RECEPTORS)	rad	DIRECTION	Beam position offset (on sky but in antenna reference frame).
PHASED_FEED_ID	Int			Phased feed
POLARIZATION_TYPE	String (NUM_RECEPTORS)			Type of polarization to which a given RECEPTOR responds. Probably R, L or X, Y.
POL_RESPONSE	Complex (NUM_RECEPTORS, NUM_RECEPTORS)			D-matrix i.e. leakage between two receptors
POSITION	Double(3)	m	POSITION	Position of feed relative to feed reference position for this antenna
RECEPTOR_ANGLE	Double (NUM_RECEPTORS)	rad		The reference angle for polarization. Converts into Parallactic angle in the Sky domain.

Notes:

FEED A FEED is e.g. a single horn. A FEED can have one or two RECEPTORs that respond to different polarization states. Feeds are numbered from 0 on each separate antenna and for each SPECTRAL_WINDOW_ID. So FEED_ID should be non-zero only in the case of feed-arrays, i.e., multiple simultaneous beams on the sky at the same frequency and polarization.

SPECTRAL_WINDOW_ID A value of -1 indicates the row is valid for all spectral windows.

BEAM_ID Points to an optional BEAM subtable with parameters (or image) for the primary beam for this antenna with this feed. No format specified yet for the BEAM table.

4.5 FIELD: Field positions for each source

FIELD: Field positions for each source				
Name	Format	Units	Measure	Comments
Columns				
Keys				
FIELD_ID	Int			Field Id
Data				
CODE	String			Special characteristics of field
DELAY_DIR	Double(2)	rad	DIRECTION	Direction of delay center.
DELAY_DIR_RATE	Double(2)	rad/s		Rate of change of direction of delay direction
NAME	String			Name of field
PHASE_DIR	Double(2)	rad	DIRECTION	Phase center.
PHASE_DIR_RATE	Double(2)	rad/s		Rate of change of direction of phase center.
POINTING_DIR	Double(2)	rad	DIRECTION	Pointing center
POINTING_DIR_RATE	Double(2)	rad/s		Rate of change of direction of pointing center.
REFERENCE_DIR	Double(2)	rad	DIRECTION	Reference center
REFERENCE_DIR_RATE	Double(2)	rad/s		Rate of change of direction of reference center.
SOURCE_ID	Int			Index in Source table
TIME	Double	s	EPOCH	Time origin for the directions and rates

Notes:

SOURCE_ID Points to an entry in the optional SOURCE subtable, a value of -1 indicates there is no corresponding source defined.

4.6 OBSERVATION: Observation information

OBSERVATION: Observation information				
Name	Format	Units	Measure	Comments
Columns				
Data				
CORR_SCHEDULE	String(*)			Correlator script
OBSERVER	String			Name of observer(s)
OBS_SCHEDULE	String(*)			Project Schedule
PROJECT	String			Project identification string.
Flags				

Notes:

OBSERVATION This contains information about the observation process. In the first instance, this would be a record only. Eventually we may want to be able to interpret Schedules for processing hints e.g. CALIBRATION_GROUPs. The OBSERVATION_ID is the row number.

4.7 OBS_LOG: Observation log information

OBS_LOG: Observation log information				
Name	Format	Units	Measure	Comments
Columns				
Keys				
OBSERVATION_ID	Int			Points to OBSERVATION table
TIME	Double	s	EPOCH	Timestamp for message
Data				
MESSAGE	String			log message

4.8 SOURCE: Source information

SOURCE: Source information				
Name	Format	Units	Measure	Comments
Columns				
Keys				
INTERVAL	Double	s		Interval
SOURCE_ID	Int			Source id
SPECTRAL_WINDOW_ID	Int			Spectral Window id
TIME	Double	s	EPOCH	Midpoint of time for which this set of parameters is accurate
Data				
CALIBRATION_GROUP	Int			# grouping for calibration purpose
CODE	String			Special characteristics of source, e.g. Bandpass calibrator
DIRECTION	Double(2)	rad	DIRECTION	Direction (e.g. RA, DEC)
NAME	String			Name of source as given during observations
POSITION	Double(3)	m	POSITION	Position (e.g. for solar system objects)
PROPER_MOTION	Double(2)	rad/s		Proper motion
SYSVEL	Double	m/s	RADIAL VELOCITY	Systemic velocity at reference

4.9 SPECTRAL_WINDOW: Spectral window setups

SPECTRAL_WINDOW: Spectral window setups				
Name	Format	Units	Measure	Comments
Columns				
Data description columns				
CORR_TYPE	Int(NUM_CORR)			Polarization of correlation
CORR_PRODUCT	Int(2, NUM_CORR)			(see below)
IF_CONV_CHAIN	Int			The IF conversion chain
NUM_CHAN	Int			# spectral channels
NUM_CORR	Int			# correlations
Data				
CHAN_FREQ	Double(NUM_CHAN)	Hz	FREQUENCY	Center frequencies for each channel in the data matrix.
MOLECULE	String			Molecule observed (optional)
REF_FREQUENCY	Double	Hz	FREQUENCY	The reference frequency.
RESOLUTION	Double(NUM_CHAN)	Hz		The effective noise bandwidth of spectral channels
TOTAL_BANDWIDTH	Double	Hz		total bandwidth for this window
TRANSITION	String			Transition of molecule (optional)
REST_FREQUENCY	Double	Hz	FREQUENCY	The rest frequency (spectral line)

Notes:

CHAN_FREQ Can be non-linear to allow for e.g. AOS

CORR_PRODUCT Pair of integers for each correlation product, specifying the receptors from which the signal originated. To get the meaning of the values (0 or 1), see the POLARIZATION_TYPE in the FEED table. An example would be (0,0), (0,1), (1,0), (1,1) for all correlations between whatever the receptors on FEED1 and FEED2 are measuring

CORR_TYPE An integer for each correlation product indicating the Stokes type as defined in the Stokes class enumeration.

IF_CONV_CHAIN Identification of the electronic signal path for the case of multiple (simultaneous) IFs. (e.g. VLA: AC=0, BD=1, ATCA: Freq1=0, Freq2=1)

NUM_CORR RR would be 1, RR, LL would be 2, XX,YY,XY,YX would be 4, etc.

RESOLUTION The effective noise bandwidth of spectral channels (the frequency resolution of each channel), this may differ from the channel spacing (difference in frequency between the centers of adjacent channels). The Vector nature allows for variable-width channels.

4.10 SYSCAL: System calibration

SYSCAL: System calibration				
Name	Format	Units	Measure	Comments
Columns				
Keys				
ANTENNA_ID	Int			Antenna id
ARRAY_ID	Int			Array id
FEED_ID	Int			Feed id
INTERVAL	Double	s		Interval
SPECTRAL_WINDOW_ID	Int			Spectral window id
TIME	Double	s	EPOCH	Midpoint of time for which this set of parameters is accurate
Data description				
NUM_RECEPTORS	Int			# receptors on this feed (1 or 2)
Data				
PHASE_DIFF	Float	rad		Phase difference between receptor 2 and receptor 1
TCAL	Float (NUM_RECEPTORS)	K		Calibration temp
TRX	Float (NUM_RECEPTORS)	K		Receiver temperature
TSYS	Float (NUM_RECEPTORS)	K		System temp
TCAL_SPECTRUM	Float (NUM_RECEPTORS, NUM_CHAN)	K		Calibration temp (optional)
TRX_SPECTRUM	Float (NUM_RECEPTORS, NUM_CHAN)	K		Receiver temperature (optional)
TSYS_SPECTRUM	Float (NUM_RECEPTORS, NUM_CHAN)	K		System temp (optional)
Flags				
PHASE_DIFF_FLAG	Bool			Flag for PHASE_DIFF
TCAL_FLAG	Bool			Flag for TCAL
TRX_FLAG	Bool			Flag for TRX
TSYS_FLAG	Bool			Flag for TSYS

4.11 WEATHER: weather station information

WEATHER: weather station information				
Name	Format	Units	Measure	Comments
Columns				
Keys				
ANTENNA_ID	Int			Antenna number
ARRAY_ID	Int			Array id
INTERVAL	Double	s		Interval over which data is relevant
TIME	Double	s	EPOCH	
Data				
H2O	Float	m^{-2}		Average column density of water
IONOS_ELECTRON	Float	m^{-2}		Average column density of electrons
PRESSURE	Float	Pa		Ambient atmospheric pressure
REL_HUMIDITY	Float			Ambient relative humidity
TEMPERATURE	Float	K		Ambient Air Temperature for an antenna
WIND_DIRECTION	Float	rad		Average wind direction
WIND_SPEED	Float	m/s		Average wind speed